**주 간 회 의 록**

|  |  |  |  |
| --- | --- | --- | --- |
| **팀 명** | Sky Rescuers | **차수** | 4차 |
| **일 시** | 2023년 4월 16일 일요일 20시 00분 | | |
| **장 소** | 창업보육센터 C동 102호 | | |
| **참석자** | 이재성, 이선호, 이지훈, 정승진, 한규영 | | |
| **불참자** |  | | |
| **주요안건** |  | | |
| **회의내용** | * **지난주 진행상황**  1. **드론 랜딩기어 형상 보완** 2. **파이썬 임베딩 코드 작성** 3. **드론 텔레메트리, 차량 텔레메트리로 통신 연결** 4. **드론 라즈베리파이에서 pixhawk로 명령보내는 함수 작성**  * **이번주 진행상황 (진행상황을 나타내는 사진 등 첨부)**  1. **드론 납땜 및 조립** 2. **미션플래너 Navigation, Do 명령어 정리 및 각각 파라미터 설정 후 데이터 포맷 작성** 3. **속도조절 알고리즘, next stoppoint설정 알고리즘, 통신오류에 대한 알고리즘 작성** 4. **Thread간 concurrent queue 연결** 5. **python코드를 임베딩하는 Makefile 수정** 6. **Linux환경에서의 텔레메트리 통신 연결**  * **지난주 피드백**  1. **간헐적 통신 오류가 있는 데이터를 활용해 차량의 위치를 큰 오차 없이 추정하는 코드를 작성하고 시뮬레이션 결과를 보일 것** 2. **드론 payload의 배치에 따른 무게중심의 변화를 보이고 안정성에 미치는 영향을 연구할 것(수식으로 표현해볼 것)**  * **문제점**  1. **전역함수를 사용하지 않고 실행함수를 사용하였을 때 init값이 저장되어야 하지만, 계속 업데이트됨.** 2. **Linux 환경에서 텔레메트리 통신 도중 재연결을 할 경우, 통신이 되지 않다가 프로세스를 종료 후 실행할 시 원활하게 작동함.** 3. **텔레메트리 통신에서 string 문자열을 전송할 경우 50자이하로만 전송할 수 있다는 문제점이 발생함.**  * **해결방안**  1. **Flag 파라미터를 이용하여 init값이 업데이트 되지 않도록 해결함.** 2. **String 문자열 전송 시 개발환경에서 50자를 기준으로 하여 나눠서 전송하고 드론에게 명령하기 전 합쳐야함.**  * **팀원 별 실천사항(각 팀원에 부여된 역할을 명확히 명시할 것)**   **이재성 – Thread concurrent queue 작성 및 Makefile 수정**  **이선호 – 차량알고리즘 (속도조절, next stoppoint 설정, 통신오류)작성**  **이지훈 – Missionplanner plan 명령어 정리, 명령어 파라미터 설정 및 데이터 포맷화**  **정승진 – Linux 환경에서의 텔레메트리 통신 구현**  **한규영 – 드론 제작 및 CATIA를 활용한 무게중심 계산**    그림 1 드론의 무게중심 설정    그림 2 Missionplanner의 명령어 목록 정리          그림 3 Missionplanner에서 도로에서의 waypoint와 stoppoint를 지정하여 설정 및 데이터 포맷화 작업  (point / 위도 / 경도 / 고도 / count\_point / Isend / nxt)    **그림 4 텔레메트리 전송 그림 5 텔레메트리 수신**    그림6 1 Thread1에서 queue의 데이터 push    그림7 Thread2에서 queue의 데이터 pop    그림8 Thread1,2의 queue 실행 결과    그림9 Makefile에서 object 파일 생성    그림10 object 파일 생성 결과    **그림11 메시지 decode**    **그림12 메시지 decode값의 결과 & 차량 데이터에 따른 드론 속도변화 출력**   * **다음 모임 시간/장소: 2023년 4월 13일 목** | | |